

6. SOP 元件

6.1 说明

Surface 元件，亦称 SOP，在整个元件大家庭中负责处理所有与三维有关的操作。

它包含简单的三维几何体，粒子系统，建筑模型，三维字体等等。因为学习难度稍大以至于 SOP 经常被许多初学者忽视。但是请放心坚实的 SOP 操作知识将给你带来许多意外的惊喜，项目中的灵感，以及提供许多非常有效的解决问题的方法。

许多项目涉及到投影映射，实时三维运动捕捉，建筑表面 LED 空间分布，视频分层播放，而如果没有 SOP 元件这将是非常困难或者不可实现。

TouchDesigner 088 目前所支持的 3D 文件类型有：

- .fbx
- .obj
- .3ds
- .dxf
- .dae

保持元件的渲染流畅度是至关重要的，这将在本书的“优化”章节做进一步的讨论。而当它来自 SOP 时，这一点显得尤为重要。通常的操作是将变换的动画数据给到 Geometry COMP 元件，而不是直接给到一个 SOP 元件。因为 SOP 变换是在 CPU 上执行，并且会将几何体上的每个点都执行一次。而组件级别的变换会将三维几何体或对象视作一个整体，在 GPU 上只运行一次。在 GPU 上的一个单独的操作与在 CPU 上成百上千次的操作相比，前者当然更有效率。

模型上的点数，基元数，顶点数与网格数取决于正在被执行操作模型，但基本原则都是模型的顶点或多边形面数越多，电脑操作时需要的运算能力与分配的 GPU 内存也越大。对于复杂模型 TouchDesigner 有内置工具可以减少多边形面数，但是使用专业的模型工具优化几何体会提升更大的灵活性。

6.2 渲染

许多初学者努力找到一条快速有效的捷径从三维工作流程过渡到二维工作流程。网上有大量的三维工作流程教程可以充分解释三维场景渲染的原理和特点，但是这一章的重点是演示如何将简单的几何对象渲染到 Render TOP。

一个三维场景由三部分组成：

1. 三维几何体（包含材质）
2. 摄像机
3. 照明

打开案例“Rendering_1.toe”。快速浏览，看到需要被渲染到场景的三个元件：摄像机，灯光与三维几何体，都被 Render TOP 所引用。我们要对它们逐一介绍，然后观察如何将它们连在一起。

让我们从三维几何体开始，因为它是最基础的。三维模型可以是简单的多边形，三维动画人物，建筑模型等等。无论导入还是生成模型，所有的操作都使用 SOP，并且这些 SOP 以 Geometry 组件结束。在 TouchDesigner 中，SOP 自身从不直接被渲染，被渲染的是包含持有渲染标记 SOP 的 Geometry 组件或 Geometry COMP。通过下面两个案例帮助理解这个重要的知识点。

打开“Rendering_2.toe”。这个项目将一个简单几何体发送给四个有着不同位置的 Geometry COMP。出于案例演示的目的，这个案例使用的是一个 Box SOP，但是它可以是其他更加复杂的模型，这种复杂的模型还可以加上一些动画和交互性。让我们回顾一下知识点：SOP 不能直接渲染，它只在逻辑上将可以容纳迭代的模型载入一次，然后在 Geometry COMP 上渲染。

打开“Rendering_3.toe”。这与前面的案例不同，它是一个单独的 Geometry COMP 中包含有被渲染的三个不同的模型。记住这点：Geometry COMP 是被渲染的，它将其元件内要被渲染的模型视为一个组。现在看起来这并不是一个很重要的行为，但是随着项目越来越复杂，节约资源将会变得非常重要。

到现在为止，各种几何体都没有材质。材质可以将三维场景变得有趣。使用水泥做的方块与普通盒子之间的区别就是材质。纹理可以被应用于两个级别中：SOP 级别与 Component 级别。第一种方法是使用 Material SOP，第二种方法是在 Geometry COMP 的“Render”面板引用材质。在案例“Rendering_4.toe”，虽然材质看起来是一样，但是它们使用不同的纹理映射方法。

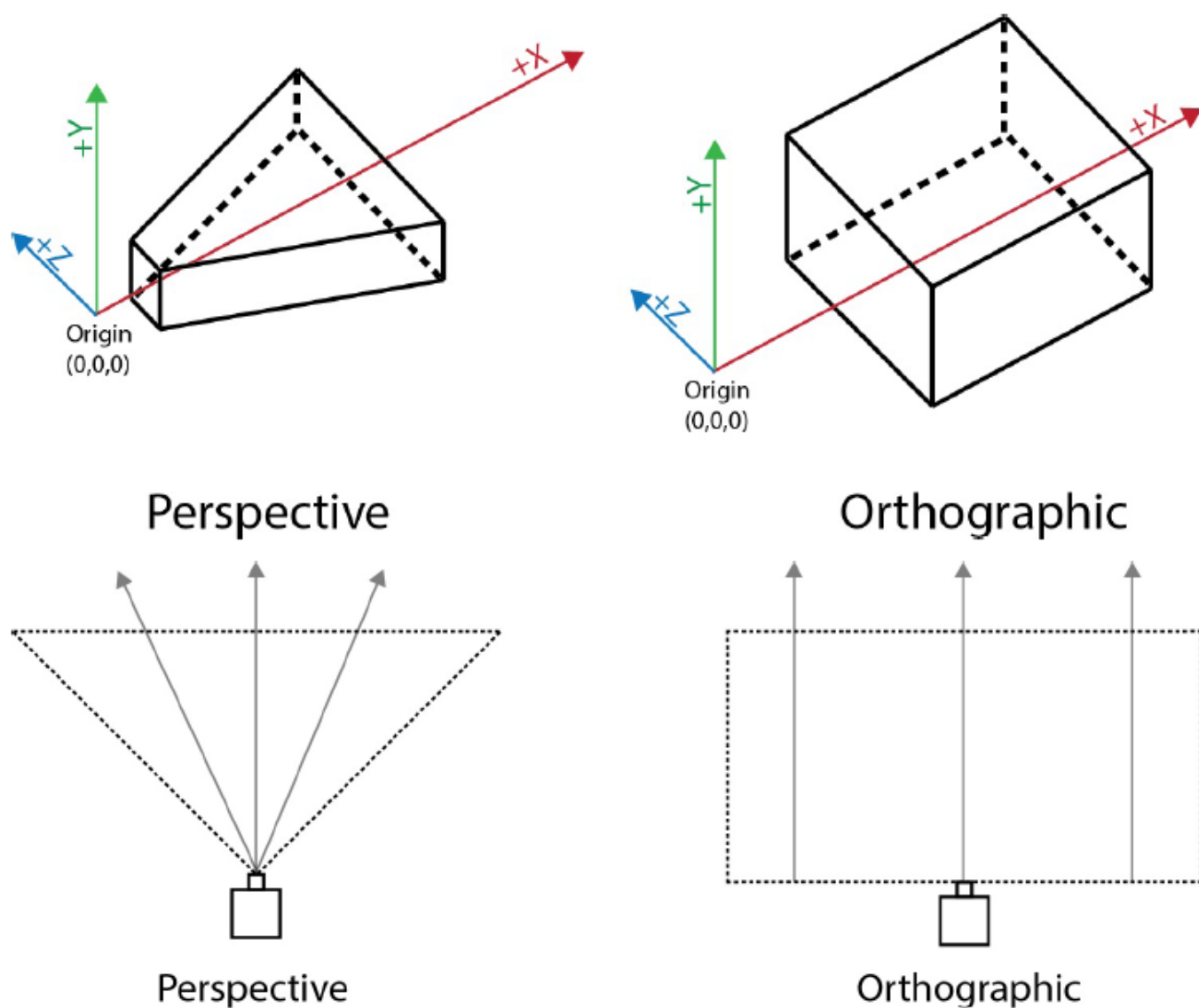
现在，材质被添加到以上的模型中了，现在让我们快速讨论一下 UV 贴图。打开“Rendering_5.toe”。这个案例与上一个相同的案例，但相同的纹理其效果看起来完全不同。这是因为 UV 贴图坐标被改变。

三维场景的第二个部分是灯光。就如现实生活一样，三维场景需要照明。打开案例“Rendering_6.toe”。在这个案例我们看到一个简单的盒子被渲染，但是在 Rendering TOP 中却没有可见的部分。这是为了通过将光线的明暗（dimness）被设置为 0 以说明照明的重要性，并且它是如何经常被忽视的。

接下来三个案例 “Rendering_7.toe” , “Rendering_8.toe” , “Rendering_9.toe” 是相同场景中使用不同的灯光。在每一个项目中, Light COMP 从不同的角度对 Box SOP 进行照明。

渲染三维场景的最后部分是摄像机。摄像机是眼睛和视角。摄像机看到什么就会渲染什么。打开案例 “Rendering_10.toe” 。整个场景中的运动都来源于摄像机的运动。通常, 当我们 将摄像机放置好之后, 就把它忘记了。这会导致一个静态, 没有活力的枯燥场景。不要害怕 像一个摄影师一样思考, 尝试改变摄像机位置, 焦距与摄像机运动。

有两种摄像机类型: 透视摄像机与正交摄像机。



透视摄像机不用多解释, 它的工作原理如人眼。透视摄像机由透视点与视锥用于决定摄像机看到的画面将被如何呈现。对于透视摄像机, 整个的三维场景都像是集中到一个点。透视校正, 亦称 “近大远小”, 被应用于场景中的所有对象, 即对象距离摄像机越远看起来越小。这种摄像机最经常被应用于模型人眼。在案例文件 “Camera_1.toe” 有这种类型的摄像机。

这个案例重点展示透视效果。有两个相同的立方体。在场景中放置较远的立方体显示较小，如现实中看到的一样。这是透视摄像机功能的最简单演示案例。

正交摄像机与透视摄像机有很大不同。那些熟悉的计算机辅助设计（CAD）软件的朋友，在这之前可能已经用过正交摄像机。正交摄像机的核心原理是没有透视点。三维场景中的所有灯光都不会像在透视摄像机中那样有一个唯一的透视点。对象不会在它们的 Z 轴上发生变形，意味着无论它与摄像机的距离如何改变，它都不可能看上去比场景中的其他对象更大或更小。

打开案例“Camera_2.toe”，这个场景与“Camera_1.toe”完全一样，但是最大的不同是渲染画面，无论两个立方体在 Z 空间如何定位，它们都并排显示。第一次接触会比较难以理解。那么就跟随下面两种思维方式：

第一种，一个三维世界在被渲染前被挤压到一个单独的平面上。想象一下超级马里奥游戏其实是一个三维场景，在渲染时使用正交摄像机。不管坏人与平台在屏幕的边缘还是中心，它们没有透视关系，总是拥有相同的尺寸与形状。

第二种，思考在 CAD 程序中设计一个建筑的蓝图。它是在二维平面上表示一个三维对象。无论蓝图显示在屏幕的哪个位置，在边缘测量的一个单位长度与在屏幕中间测量的一个单位的长度完全相同。摄像机位置的任何变化都不会变形正在渲染的画面。

这并不是说当使用正交摄像机时景深就变得无关。相反，当几何体在场景中发生层叠时，景深变得非常必要。

摄像机在投影映射项目中同样重要。投影映射将在以后的案例中探讨，但现在，关键在于理解摄像机的另一个重要角色。在投影映射中其目标是将现实对象变成三维显示表面。要做到这一点，一个对象模型需要被导入到 TouchDesigner，使用各种资源来为它添加纹理。这个纹理对象接下来需要被渲染。Camera COMP 用来模拟现实中的投影机。收集尽可能多的投影机的特征信息以及它的镜头，Camera COMP 可以在 TouchDesigner 中精确模拟投影机的视野设置（FOV）。这些视野设置将被渲染，输出，校准与 lined up 现实对象，这就是投影映射的基础。